Feature Extraction Techniques in Human Foreskin Fibroblasts Time-lapse Videos

CS 279 Final Project
Min Cheol Kim

SUnetID: mincheol

Introduction

Substantial progress has been made in characterizing the mechanisms of tension sensing, adhesion, and traction in cells on two-dimensional (2D), rigid surfaces. Mesenchymal migration in particular has been shown to involve cycles of leading edge protrusion, adhesion maturation, and translocation of the cell body. These processes depend on the mechanical forces produced by directional actin polymerization at the leading edge and myosin contraction that causes retrograde flow of actin filaments, and models have been proposed for how these components interaction to produce cell movement. However, there are inherent differences in cellular architecture, signaling, and behavior when cells are embedded in the soft and confined environment of the extracellular matrix.

One approach to studying three-dimensional cellular movement is using time-lapse videos of a cell's volume space and analyzing the result. In this project, the data used was a three-dimensional time-lapse video of Human Foreskin Fibroblasts (HFFs) embedded in a fibrin gel. These HFFs had their myosin labelled with Green Fluorescent Protein (GFP) and at some point in the video, a myosin light chain kinase (MLCK, a molecule thought to be important for cell movement and force generation) inhibitor was added to the cellular environment.

This project is a part of Leanna Owen's (a biophysics graduate student) research on characterizing actomyosin dynamics and force generation in three-dimensional mesenchymal cell movement.

Project Goal

This project aims at developing tools for feature extraction and making comparisons between cells that have not been drugged yet and cells that have been in the drug for various amounts of time.

Data and Experimental Set up

The data used for this project is a time-lapse video of 53 time-points of 54 z-slices. Each z-slice is an image of 1492 x 1018 pixels. ML7 drug, the MLCK inhibitor, was added between time-points 34 and 35. Only the cells high above the z-stack, not touching the glass, were used in this experiment to see how cells truly in a 3D environment behave. A z-projection of a specific time-point is shown in Figure 1.

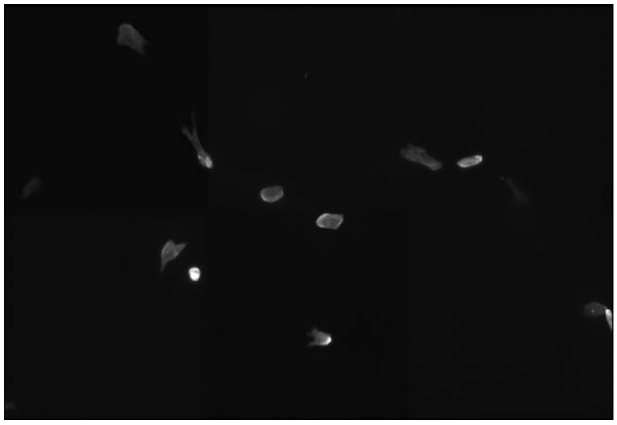

Figure 1. Sum Intensity Projection at Time-point 1

Approaches

I tried several methods for trying to see if any morphological changes occurred to the cells after the drug was administered. My first approach was to hand-pick several features and measure them frame-by-frame, attempting to see if I could see changes between pre-dreg and post-drug states. Some of the features that I chose are cell volume, cell x-y spread area (after visually noting that the cells have a tendency to spread in the x-y plane more than any other direction), and cell length. My second approach was performing principal component analysis (PCA) on the outlines of the x-y projections of the individual cells, simply using one time point in a pre-drug state and a post-drug state.

Methods

Processing and analyzing the data for this project required several steps. The first few steps of processing was the same for all of my approaches.

Creating a binary mask
First, a binary mask of the volume time series was created using ImageJ using the triangle method.

Mini-volume generation
The next step was isolating individual cells in each time frame so that their cellular characteristics can be examined. Figure 2 shows a representative diagram of this process. To do this, the individual cells were labelled using the bwlabel() function in MATLAB and the smallest cube that contains the entire cell was cut out from the volume and was saved as a new, smaller volume. This process created 53 folders (each for one time point) with a certain number of cells in each folder.

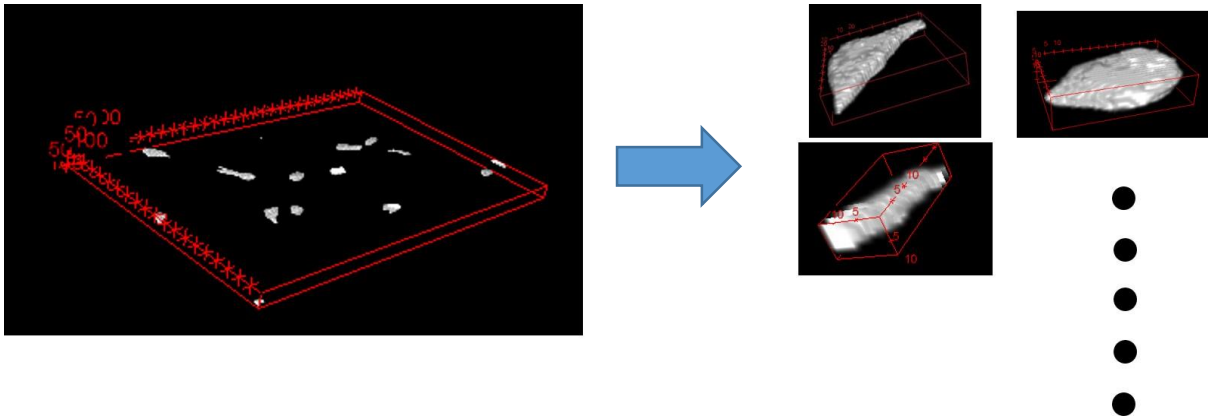

Figure 2. Creating mini-volumes

From here on, different manipulations were used depending on what approach I took, and what feature I was looking for.

Approach #1: Handpick few features and measure them across time

Measuring cell volume

This approach was relatively simple and only involved counting the number of white-voxels (value of 255) that each cell spanned. Using this approach, a set of numbers representing the cell volumes was calculated for each time-point, and they were represented in a boxplot in Figure 3. The raw data, before the boxplot generation, is shown in Figure 3. In the raw data, where each point represents a single cell volume, is shown in Figure 4. In these plots and ones following, the orange line indicates the time of drug addition.

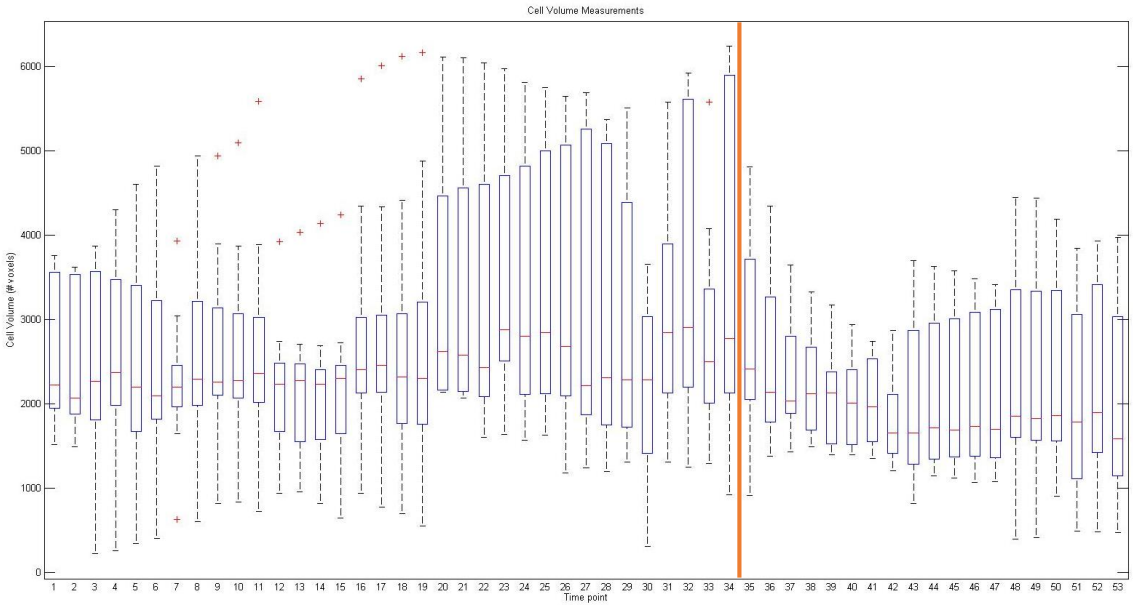

Figure 3. Cell volume boxplot per time-point

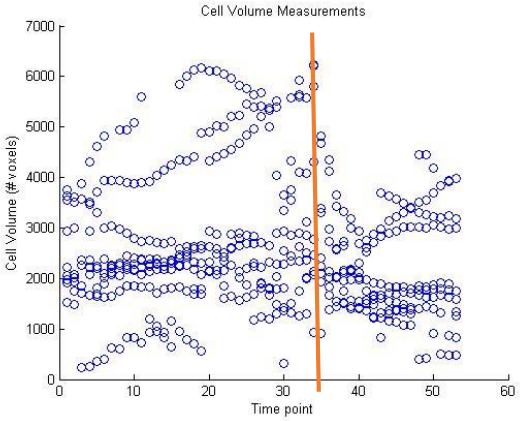

Figure 4. Cell volume raw data per time-point

Measuring cell x-y spread area

We have noticed that HFFs tend to spread in the x-y plane even when it is high up in the gel and is not touching the glass. The reason for this isn't clear, but Leanna and I thought it might be a good measure of cell dynamics to make an x-y projection of these cells and measure the x-y area that the cell spans. A representative diagram for this projection is shown in Figure 5.

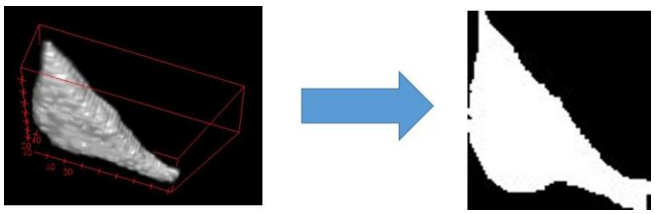

Figure 5. Making x-y projections

As with measuring cell volume, a plot of raw data (again, each point representing the x-y area of a single cell) and the boxplot per time-point was generated and is shown in Figure 6 and 7.

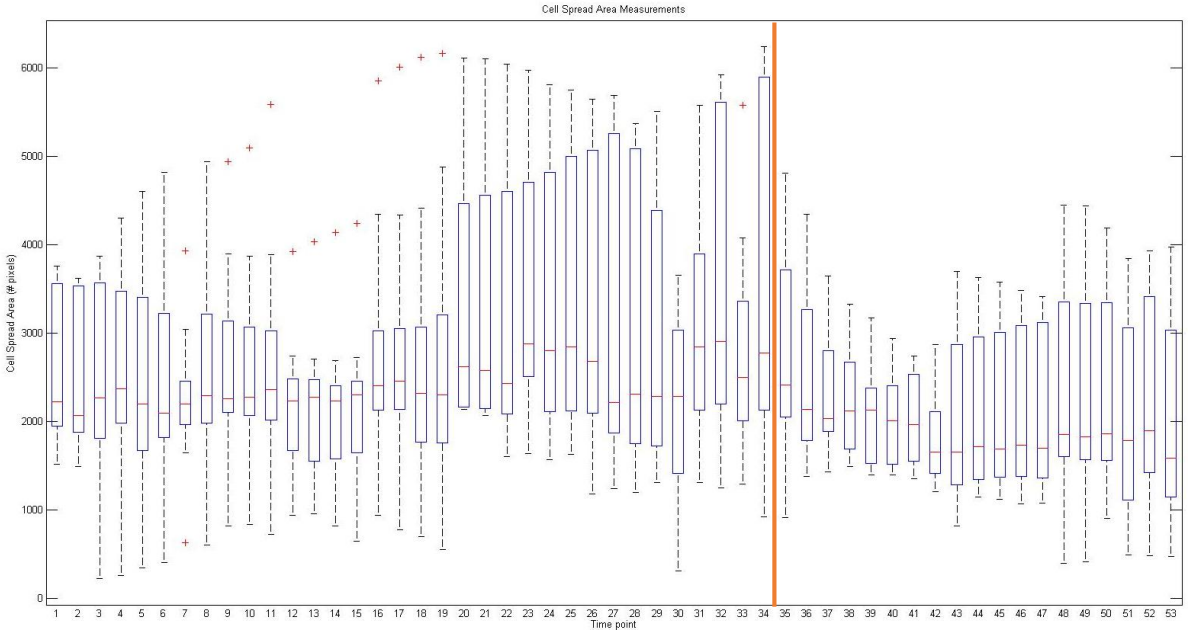

Figure 6. Cell spread area boxplot per time-point

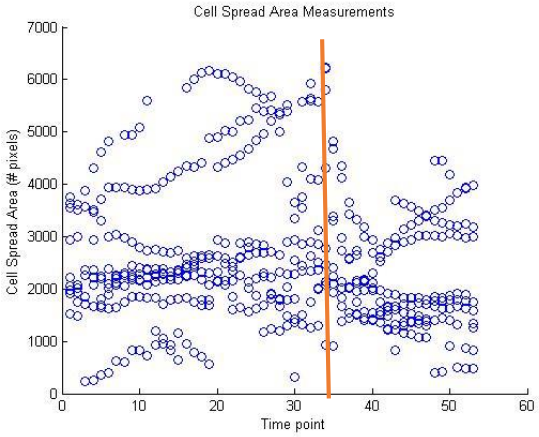

Figure 7. Cell spread area raw data per time-point

Measuring cell length

Another measure I decided to go with was cell length, which I estimated to be the diagonal length of the smallest cube that can contain the cell. Similar boxplots and scatter plots were generated and is shown in Figures 8 and 9.

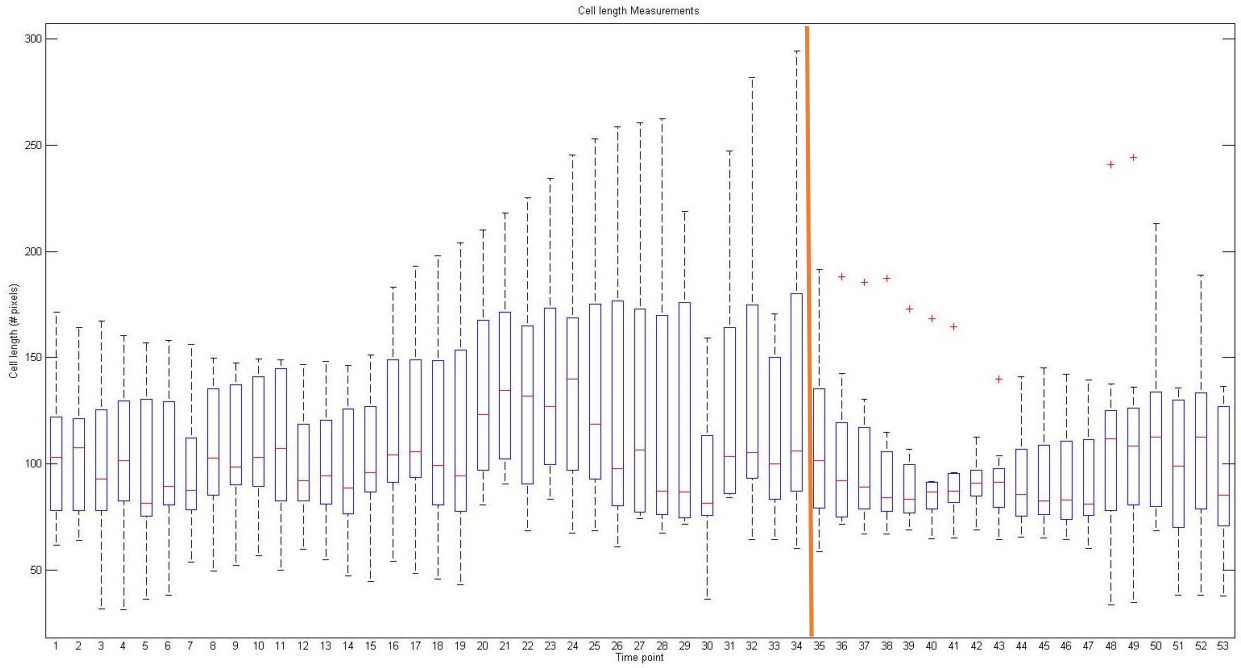

Figure 8. Cell length boxplot per time-point

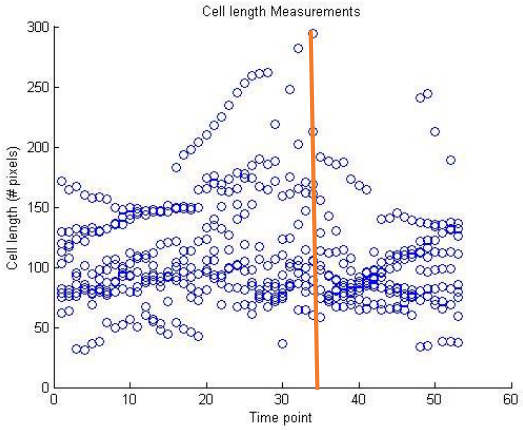

Figure 9. Cell length raw data per time-point

Approach #2: Principal Component Analysis of 2D Cell Outlines

The second approach was using principal component analysis (PCA) on the outline of a x-y projection of a cell. For PCA, I only used cells from two time-points: 1 and 50. I felt that performing PCA on every cell in every time-point would simply learn the features specific to these cells, and I also felt that time-point 50 was sufficiently after the drug took its effect. This approach took several steps with varying successes.

Generation cell outlines

To generate the outlines, I first made x-y projections of the cells (much like I did when measuring cell spread area). From the x-y projection, I used the bwboundaries() function in MATLAB to extract the outline of the binary image. The process is shown in Figure 10.

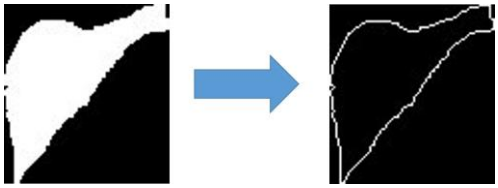
Figure 10. Cell outline generation

Sampling points

To perform PCA on the points, we need to sample the same number of points from each outline. To do this, I found the smallest (one containing the smallest number of pixels) cell boundary and used that as the number of samples. That number of points was randomly chosen from every other outline. This sample selection process is shown in Figure 11.

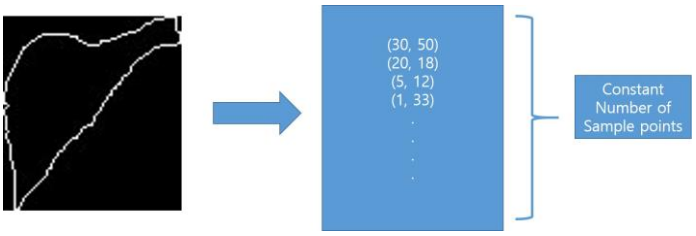

Figure 11. Sampling points from an outline

Alignment and Procrustes Analysis

To align the cells, I used the procustes() function in MATLAB on the sample points for different cells, finding both the transformation matrix and the transformed outline. To make sure that all the cells are aligned the right way, I thought it would make sense to first align all cells to the first cell, then align all cells to the next cell, until every cell has been aligned to every other cell. I tried repeating this process multiple times in an outer loop, but it did not have much effect. An attempt to outline the cells is shown in Figure 12.

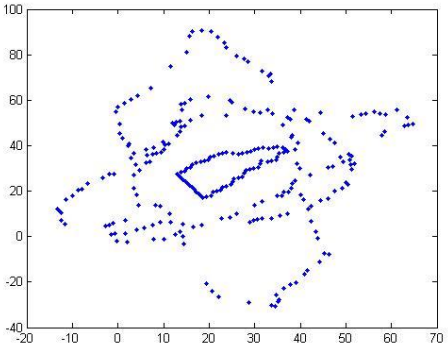

Figure 12. Outline alignment attempt with Procrustes analysis

Principal Component Analysis

Using the cell outlined by the Procustes analysis, I performed PCA using the MATLAB PCA() function. I found that the first two components account for around 70% of the variation in the sample points for each cell. Because the sample selection was random, PCA yielded some different results each run. Two such plots of the first and second principal components are shown in Figure 13. In these plots, the different colored dots represent cells from different time-points (pre-drug condition and post-drug condition).

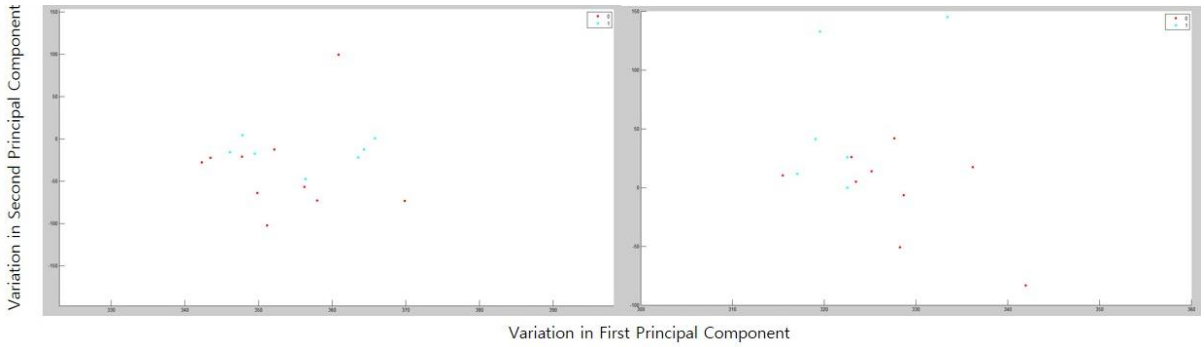

Figure 13. Examples of graphs generated by PCA

Discussion

The first approach that I took, measuring cell volume, spread area, and length, seemed to at least somewhat capture the effect of the drug added at time-point 34. All three measurements seems to fluctuate until they visually decrease after the addition of the drug. A more rigorous statistical treatment of the boxplots/data would be better, but my time scope for the project

didn’t allow for that. The decrease in these features could be representative of the cell at least partly losing its ability to generate force and moving forward, or losing the ability to maintain an adhesion with the fibrin along its protrusions.

The PCA did not really seem to separate the two groups. The first aspect of my process that makes PCA ineffective would be random sampling of the cell outline. For the PCA to be the most meaningful, I would have to sample points in the outline in such a way that the distance between neighboring points stayed constant; this was not maintained with random sampling. Second, a better alignment strategy, besides iteratively performing Procrustes transformations, might make the PCA more meaningful. Third, I believe that simply having more cells (perhaps from more videos) to perform PCA on would improve the results.

Future work

The goal of the project was to develop a pipeline for analyzing these videos and even though some approaches did not work out as hoped, it served as a good experience for getting to know these tools more and getting more ideas. My future work would involve trying to fix some of the problems with the PCA mentioned above, while trying to see if other methods of representing shape works better. One idea might be to change the cell shapes into [x, y] curves transforming the curves into a Fourier series in the spatial frequency domain, and seeing if the coefficients show any differences.

Acknowledgements

Many thanks to Leanna Owen in the Dunn Lab (Chemical Engineering) for providing me with the time-lapse video, and for being such a great mentor in general!